



ثانوية التكنولوجيا التطبيقية  
Applied Technology High School

---

## Microcontrollers

### Module 1: Introduction to Microcontrollers



---

PREPARED BY

**Academic Services Unit**

April 2012

© Applied Technology High Schools, 2012



# Module 1: Introduction to Microcontrollers

## Module Objectives

Upon successful completion of this module, students will be able to:

- Name some gadgets that use microcontrollers.
- Define a microcontroller.
- Differentiate between a computer and a microcontroller.
- Describe the components of a microcontroller.
- Define the pin functions of the BASIC Stamp 2 (BS 2).
- Name the parts of the BASIC Stamp Board of Education (BoE).
- Describe the function of each part of the BASIC Stamp BoE.
- Identify and name the three main panels of the BASIC stamp Editor.
- Demonstrate the function of the DEBUG command, DEBUG formatters and control characters by running simple BS2 programs.

## Module Contents:

	Topic	Page No.
1.1	Introduction	2
1.2	BASIC Stamp 2 Microcontroller	4
1.3	Lab Activity 1	8
1.4	Lab Activity 2	10
1.5	Lab Activity 3	12
1.6	Review Exercise	14
1.7	Assignment	16

## 1.1 Introduction

In our daily life, we come across a number of electronic gadgets or gizmos such as clocks, radios, mobile phones, and MP3 players. If any of these devices have a button and a digital display, chances are that it also has a programmable microcontroller. If we make a list of the devices with microcontrollers that we use in a typical day, it can be an endless list. Here are some examples: if your clock alarm goes off, and you hit the snooze button a few times in the morning, the first thing you do in your day is interact with a microcontroller. Heating up some food in the microwave oven and making a call on a cell phone also involve the use of microcontrollers.

	
Cell phone	Digital Alarm Clock
	
Microwave oven	Handheld remote

Figure 1.1 Examples of various devices that use microcontrollers

Other examples include turning on a television with a handheld remote, playing a handheld game, using a calculator, checking your digital wristwatch- all those devices have microcontrollers inside them that interact with you.

**What are microcontrollers?** Microcontrollers are special purpose computers that can do one task well. In this module, you will be introduced to the BASIC Stamp 2 microcontroller developed by Parallax.

### **1.1.1 Comparison between a computer and a microcontroller**

Like a computer, a microcontroller has the following parts:

- Central Processing Unit (CPU)
- Memory
- Input and Output devices

However, unlike a computer that can run many tasks at a time, the microcontroller can do only one dedicated task. They have the following special features:

- 1) They can be embedded inside some other devices to control their actions.
- 2) They can be dedicated to one task and can run one specific program well.
- 3) They are low power devices.
- 4) They are small in size and are inexpensive.

### **1.1.2 Microcontroller Architecture**

Microcontroller is a small computer on a single IC consisting of

1. Central processing unit (CPU)
2. Memory
3. Input and Output ports (I/O pins)

1. **CPU:** The CPU is the “brain” that executes the programs. Moreover it can perform addition, subtraction, division, and multiplication operations.

2. **Memory:** The microcontroller memory is used to store programs. It consists of cells called memory locations. Just like a computer memory, a microcontroller has short term (volatile) memory and long term (non-volatile) memory.
3. **Input and Output Ports:** The microcontroller uses its input and output ports to communicate with the outside world. Through its Input/Output (I/O) ports, it is possible to connect various input and output devices to the microcontroller. Examples of input devices include switches, pushbuttons, sensors. Examples of output devices include buzzer, LED, lamp and so on.

## 1.2 BASIC Stamp 2 Microcontroller

Parallax, Inc.'s BASIC Stamp 2 module has a microcontroller built onto it. It is a black chip with lettering on it that reads "PIC16C57". The rest of the components on the BASIC Stamp module are also found in consumer appliances you use everyday. All together, they are correctly called an embedded system. Frequently, such modules are commonly just called "microcontrollers".

The activities in this module will guide you in building circuits similar to the ones found in consumer appliances and hi-tech gadgets. You will also write computer programs that the BASIC Stamp module will run. These programs will make the BASIC Stamp module monitor and control these circuits so that they perform useful functions.

Basic Stamp2 has 16 (I/O) ports. Each one is bi-directional; this means that each I/O port can be programmed to be either an input or an output port.

### 1.2.1 BASIC Stamp 2 Pin Assignment

Normally microcontrollers are soldered onto printed circuit boards (PCBs) and connected to various electrical and electronic components. It is very essential to know the pin assignment of the microcontroller used, so you will be able to connect the power, establish communication with PC, and connect the I/O devices to the correct input and output pins. The pin assignment/functions of BS2 microcontroller is illustrated in Figure 1.2.

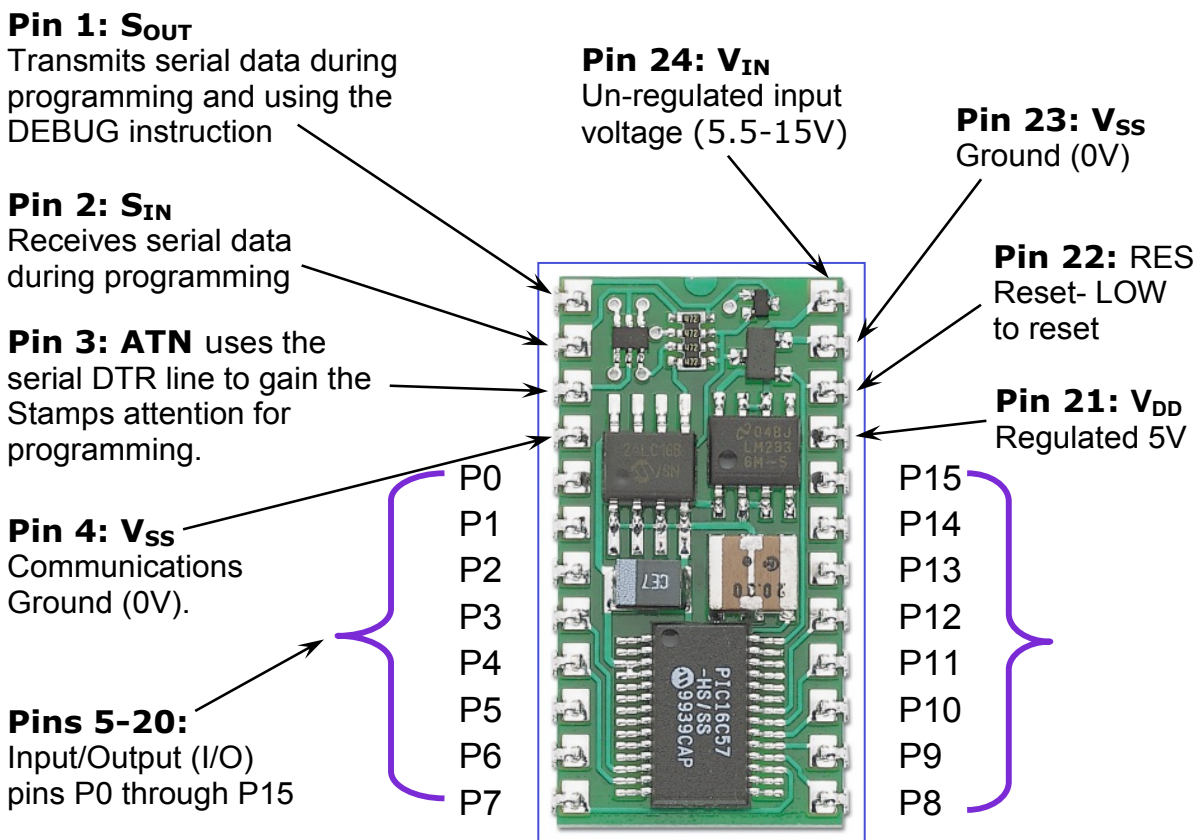


Figure 1.2: BS2 Pin Functions

### 1.2.2 BASIC Stamp Board of Education (BoE):

To power and program the BS2 module, it needs to be integrated in a PCB board. The PCB board that comes with the BS2 is called Board of Education (BoE) and is shown in Figure 1.3. This board is fitted with components and IC 24-pins Socket to plug in the BS2 and easily establish the following:

- to connect a power supply to run the BS2.
- to program the BS2 through serial or USB cable from PC.
- to build electronic circuits and control them from the BS2.
- to connect and control servo motors.

The main parts of the BoE are shown in figure 1.3.

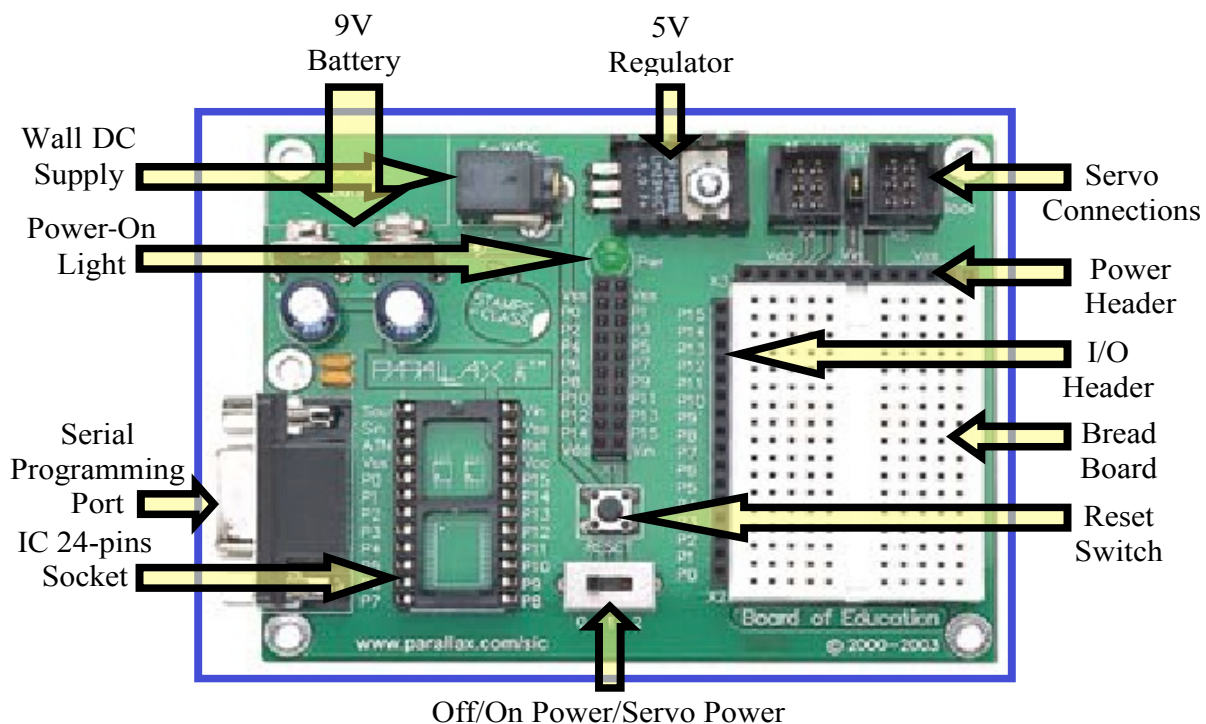


Figure 1.3: Main parts of the BoE



**Function of BoE Parts:**

<b>Part</b>	<b>Function</b>
IC 24-pins Socket	To plug in the BS microcontroller. Orientation of the microcontroller is very important.
Breadboard	Breadboard is a prototype board that allows insertion of components for practical tasks. The Breadboard is separated into rows of five sockets in two columns. The five sockets in each row are internally connected.
I/O Header	The vertical black 16-sockets labeled (P0→P15); it is used to make connection to I/O pins of the BS2, and is referred to as the I/O Header.
Power Header	The horizontal black sockets used to supply power to the circuits you build and is referred to as the Power Header.
Selector switch	Selector switch to switch ON/OFF the BoE. It has 3-positions: <b>0</b> → OFF, <b>1</b> → ON, and <b>2</b> → Servo power ON.
LED indicator	To indicate if the BoE is powered and switched ON.
Serial Programming Port	9 pin connector to connect your serial cable to PC.
Battery socket	Battery socket to connect the 9V DC battery.
Servo Connections	To connect up to 4-servo motors
Reset Switch	Allows resetting of programs. Press & release to start the program on your BS2 again
5V regulator	To regulate (5.5→15V <sub>DC</sub> ) input voltage to 5V

The Power Header is divided into 3 parts as follows:

- VDD ⇒ provides +5 Volts DC
- VSS ⇒ represents the GND (0 Volts)
- V<sub>IN</sub> ⇒ Supply Voltage from battery (9V) or wall DC supply

### 1.3 Lab Activity 1

**Objective:** To familiarize with the BASIC Stamp Editor interface.

**Background:** Microcontrollers are special purpose computers that do only one task well. Without a suitable program, it cannot handle the required task and it is of no use. Therefore every microcontroller comes with a software that can be used to program the unit. The software (PBASIC) used to program the BS2 and other BS-modules, is called *Basic Stamp Editor* (Version 2.0 or higher). The version we are going to use is v2.3.9. Besides programming, this software can display messages sent by the BASIC Stamp. This feature is very useful for troubleshooting purposes. Programming the BS2 means writing a set of commands that the microcontroller will understand and execute.

To open the BS Editor, double click the shortcut. A typical screen of BS Editor, consists of three main panels as shown in Figure 1.4.

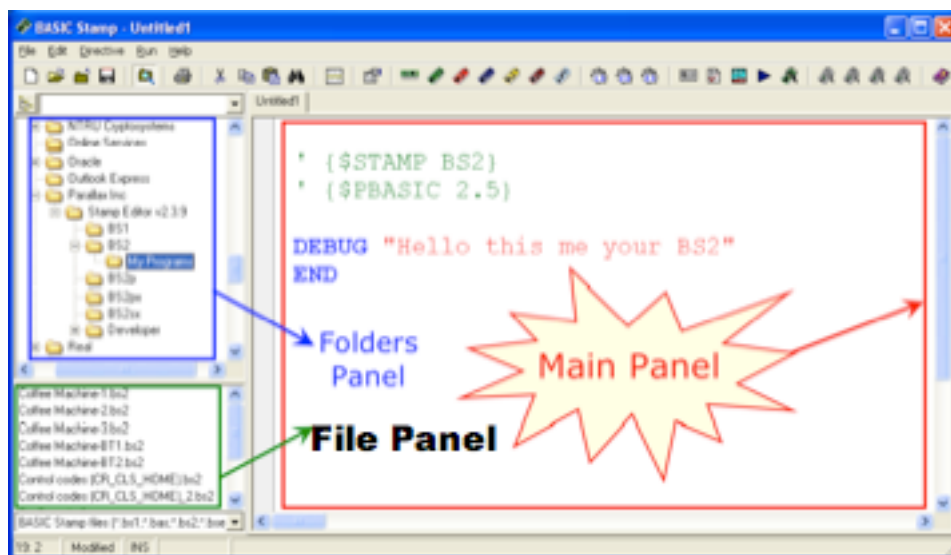
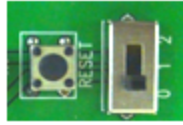

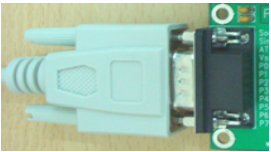





Figure 1.4: BS Editor

**Follow the 8 steps below to program the BS2:**

1. Set the 3-position switch to position-0, the BOE is OFF	
2. Plug in the 9V battery	
3. Connect your BS2 to the PC (or Laptop) through the serial (or USB) cable	
4. Open the BS Editor (double click the shortcut on the desktop)	
5. Switch ON the power of the BOE (set the SW to position-1)	
6. Identify your BS2 by clicking the ID Button	
7. Select the Basic Stamp mode ⇒ BS2 ⇒ ' {\$STAMP BS2} 8. Select the PBASIC version ⇒ Ver-2.5 ⇒ ' {\$PBASIC 2.5}	
 <p>Select by clicking on the icons (BS2 &amp; Ver-2.5)</p>	
Now the BS Editor is ready for writing a program and the BoE is ready to download and run that program.	

## 1.4 Lab Activity 2

**Objective:** To write a simple program on the BS2 editor, download and run the program.

**Background:** The first program you will write and test will tell the BASIC Stamp to send a message to your PC or laptop. The microcontroller sends a stream of ones and zeroes to communicate the text characters displayed by the PC or laptop. These ones and zeroes are called binary numbers. The BASIC Stamp Editor software has the ability to detect and display these messages as you will soon see. The "DEBUG" command will be used to write the message or display information on the debug terminal (Editor window).

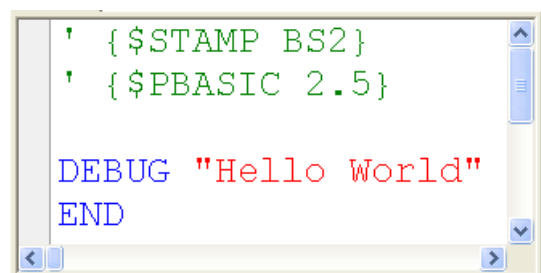
### First Program:

Your first program is a simple code where 'DEBUG' command is used to display or send information to the programming PC.

- **First:** do not forget the 8-steps to get your BS2 ready for programming.
- **Second:** enter the code of Figure 1.5 in the BS-Editor window.
- When done, **Run** the program by:
  - Clicking on (Run Menu --> select Run); or
  - Click on Play Icon (▶); or
  - Press (Ctrl + R) or Press F9 key.

You should see (Hello World) in the debug window (terminal).

The 2-lines that read **{ \$STAMP BS2 }** and **{ \$PBASIC 2.5 }** are called the **stamp directives**. They are used to indicate the kind of BS and the Version of software you are using.

A screenshot of the BASIC Stamp Editor window. The code is as follows:

```
' { $STAMP BS2 }  
' { $PBASIC 2.5 }  
  
DEBUG "Hello World"  
END
```

The code is color-coded: the directives are in green, the DEBUG command is in blue, and the string "Hello World" is in red. The window has a standard Windows-style border with a title bar and scrollbars.

Figure 1.5: My first program

The first directive is called the \$STAMP Directive, and it tells the BASIC Stamp Editor that you will be downloading the program specifically to a BASIC Stamp 2 module. The second directive is called the \$PBASIC directive, and it tells the BASIC Stamp Editor that you are using version 2.5 of the PBASIC programming language.

A command is a word you can use to tell the BASIC Stamp to do a certain job. The first command in this program is called the DEBUG command. DEBUG "Hello World"

This is the command that tells the BASIC Stamp to send a message to the PC using the serial cable.

The second command is called the END command.

END

This command is used to inform the BASIC Stamp that the program is complete. When this command is executed, the BASIC Stamp will enter the **low power** mode. In the low power mode, the BASIC Stamp will wait either for a "Reset" or for a new program to be loaded. If the Reset button on your board is pressed, the BASIC Stamp will re-run the program you loaded into it. If a new program is loaded into it, the old one is erased, and the new program begins to run.

Run the program, and observe the output on the DEBUG terminal.

What is the message displayed?

---

### 1.5 Lab Activity 3

**Objective:** To write a program using DEBUG formatters and Control Characters

**Background:** A DEBUG formatter is a code-word you can use to make the message the BASIC Stamp sends appear in a certain way in a Debug Terminal. DEC is an example of a formatter that makes the Debug Terminal display a decimal value. An example of a control character is CR, which is used to send a carriage return to the Debug Terminal. The text or numbers that come after a CR will appear on the line below the characters that come before it. You can modify your program so that it contains more DEBUG commands along with some formatters and control characters. Try the program below to see how the message appears.

#### Second Program:

- Do not forget the 8-steps to get your BS2 ready for programming.
- Save the program under a new name by clicking *File* and selecting *Save As*. Name the file "SecondProgram.bs2".
- Include the comment section in the beginning of the program as follows:  
`What's a Microcontroller – SecondProgram.bs2  
`BASIC Stamp sends messages to Debug Terminal using Debug formatters
- Add the following three lines between the comment and the END command  
DEBUG "Hello, it's me, your BASIC Stamp!"  
DEBUG CR, "What's 7 X 11?"  
DEBUG CR, "The answer is: "  
DEBUG DEC 7 \* 11
- Save the changes you made by selecting File-Save.

- Run your modified program, and note down the results below:

---

---

---

---

- What is the purpose of the comment line? Is it ignored by the BASIC Stamp? Is it meant for human reading?

---

---

---

---

### 1.6 Review Exercise

1. Define a microcontroller.

---

---

---

---

2. Is the BASIC Stamp module a microcontroller, or does it contain one?

---

---

---

---

3. What does an apostrophe at the beginning of a line of PBASIC code signify?

---

---

---

4. Explain what the asterisk (\*) does in this command

DEBUG DEC 7 \* 11

---

---

---



5. What will the Debug Terminal display if you run the command given below?

DEBUG DEC 7 + 11

---

---

---

6. There is a problem with the commands given below. When you run the code, the numbers they display are stuck together. Modify the commands so that the numbers appear on two different lines.

DEBUG DEC 5 \* 2

DEBUG DEC 5 + 2

---

---

---

7. Name the microcontroller used by the BASIC Stamp module.

---

8. What is the purpose of the \$STAMP directive?

---

---

---

## 1.7 Assignment

1. Write a BS2 program that displays the following message:

Hello

What is your name?

2. Write a BS2 program that displays the solution to the Math problem:

$1 + 2 + 3 + 4$

The sum of the numbers must be displayed as follows:

The sum of the numbers is:

10

3. Write a BASIC Stamp program to display the following message on the debug terminal.

